

REMARKS

Claims 1 – 19 are pending in this application with claims 1, 3 – 5, 6 – 8, and 10 – 17 being amended by this response. Support for the amendments can be found throughout the specification and original claims, particularly on page 9 lines 5 – 22, page 13 lines 13 – 30, and page 15 lines 1 – 21. Applicant respectfully submits that no new matter has been added by these amendments.

Objection of claims 4 and 8

Claims 4 and 8 are objected to for lack of antecedent basis.

Claims 4 and 8 have been amended to provide proper antecedent basis for the terms “browser window object” in claim 4 and “image” in claim 8. In view of the amendments to claims 4 and 8, it is respectfully submitted that this objection is satisfied and should be withdrawn.

Rejection of claims 1, 5, and 16 under 35 U.S.C. 102(b)

Claims 1, 5, and 16 are rejected under 35 U.S.C. 102(b) as being anticipated by Joseph (U.S. Patent No. 6,038,603).

Amended independent claim 1 provides a system employed by at least one web browser compatible executable application for initiating an action by a non-web enabled executable application. The system includes a web browser compatible executable application. The web browser compatible executable application employs a URL processor for generating, in response to a user command, a URL link in a predetermined particular format including context information and an action request identifier. The URL link represents an action to be performed by the non-web enabled executable application. The system further provides a URL link interpreter, coupled to the URL processor via a first mode of communication, for identifying the URL link in the predetermined particular format, for extracting the context information and action request identifier from said URL link in the predetermined particular format and initiating termination of the first mode of communication. The system also provides a communication processor, coupled to the URL link interpreter, for initiating communication of the extracted context information and action request identifier to the non-web enabled executable application by a second mode of communication different from the first mode of communication.

The present claimed system advantageously enables the execution of a **non-web enabled** application at the request of a web client on its installed host, using context information from the web client, rather than by copying executable code of the non-web enabled application to the web client host machine for execution on the web client host machine, thereby avoiding possible security problems. As the specification discusses, the system utilizes a standard web browser and a web browser compatible executable application to initiate an action by a non-web enabled application through a mediation agent, and to pass context information to the non-web enabled application (Application page 5 lines 3 – 12 and 25 – 27, and page 6 lines 12 – 23). For the reasons presented below, Applicant respectfully submits that Joseph neither discloses nor suggests each feature of amended claim 1.

Joseph neither discloses nor suggests a web browser compatible executable application employing a URL processor for generating “a URL link in a predetermined particular format including context information and an action request identifier, the URL link representing an action to be performed by a **non-web enabled** executable application” as recited in claim 1. Joseph addresses the situation in which a client requests via its communications network a resource for which the client browser has no supporting protocol (Joseph col. 3 lines 34 – 38). Employing the definition in common usage throughout the community of software engineers and developers, a protocol is the “set of conventions governing the treatment and especially the formatting of data in an electronic communications system” (Merriam Webster Collegiate Dictionary, definition 3b). Joseph describes a two-part system. The first part describes a web browser using an “encapsulating protocol” and an “operation protocol key” added to a URL. This remedies the situation in which the protocol specified as part of the original URL, which Joseph labels the “operation protocol,” is not supported by the client browser. In Joseph, the client browser is augmented with a “URL handler” and a “resource viewer” that “encapsulates” the requested URL after the URL handler determines that the requested URL specifies a non-supported operation protocol (Joseph col. 7 lines 11 – 31). Joseph merely describes pre-pending to the URL a predetermined encapsulating protocol identifier, plus a host machine designator and an operation protocol key (Joseph col. 5 line 53 – col. 6 line 14). Thus, Joseph does not disclose the operation of “generating, in response to a user command, a URL link . . . including context information and an action request identifier . . . representing an action to be performed by said non-web enabled executable application” as recited in claim 1. The encapsulated URL of Joseph is created by the URL handler after user input or other specification of the unaugmented URL, but Joseph does not contemplate adding to the URL “context information and an action identifier.” Thus, in Joseph, any data representing parameters associated with the URL will only be present if it was already part of the URL supplied by the user. Therefore, Joseph fails to contemplate generating a URL link with “context information and an action request identifier” and instead, merely converts a

previously provided link into a different format. Moreover, the Office Action correctly recognizes in paragraph 22 that “Joseph doesn’t disclose wherein said second executable application is a non-web enabled application” such as is claimed in amended claim 1.

The second aspect of the system in Joseph, residing on the server that receives the encapsulated URL, comprises a “URL broker” that is able to identify the appropriate protocol handler for the received URL. Joseph further provides a set of “resource programs” that handle at least some of the requests by optionally requesting authentication information from the original client browser, sending a request to a second server (identified by the pre-pended host machine designator) to retrieve the requested resource, and then returning the requested resource to the original client via an acceptable protocol (Joseph col. 8 lines 25 – 29). Unlike the present claimed system, the second application in Joseph is merely an application for which the client browser does not include the appropriate supporting protocol. Joseph requires that the server receiving the encapsulated URL be able to supply the appropriate protocol for communicating with the second application. After having sent the encapsulated URL to a server, Joseph merely mentions either executing “a resource program” or “retrieving a resource” in response to receiving a URL with an encapsulating protocol (Joseph, col. 10 lines 42 – 46). In contrast to the present claimed system’s support for a non-web enabled application, Joseph states that the resource program will logically, and possibly physically, access another server to retrieve the desired resource or response (Joseph col. 10 lines 58 – 67). Thus, Joseph is only able to function in an entirely web-enabled environment and does not contemplate the control or interaction with a “non-web enabled executable application”.

Furthermore, unlike the present claimed system, Joseph neither discloses nor suggests providing a feature for “initiating termination of the first mode of communication” and “initiating communication of said extracted context information and action request identifier to said **non-web enabled** executable application by a second mode of communication” as recited in claim 1. The section of Joseph relied on in the Office Action, Joseph col. 3 lines 64 – 67, makes no mention or suggestion of “initiating termination of the first mode of communication” as recited in claim 1 of the present system, because Joseph does not contemplate that the URL refers to a non-web enabled application as in the present claimed system. The “URL handler” and set of “resource viewers” described by Joseph merely support issuing requests for resources “over the communications network” (Joseph, col. 7 line 67). It is clear that, in contrast with the present system, Joseph does not disclose or contemplate communication with “a non-web enabled executable application by a second mode of communication different from said first mode of communication” as recited in claim 1. Consequently, Applicant respectfully requests the withdrawal of the rejection of claim 1.

Claim 5 is dependent on claim 1 and is considered patentable for the same reasons presented above with respect to claim 1. Claim 5 further provides that the context information includes at least one of a user identifier, a patient identifier, a customer identifier, a source identifier, a destination identifier, a password, a computer operational session identifier, an identifier identifying a non-web enabled executable application to perform the user command, and a data identifier; and the action request identifier identifies at least one of a type of action to be performed by the second, non-web enabled application, a type of user command, an event, and an authentication service. Claim 5 is further considered patentable because Joseph col. 4 lines 6 – 12 or elsewhere contains no mention or suggestion of context information comprising any one of “a user identifier, a patient identifier, a customer identifier, a source identifier, a destination identifier, a password, a computer operational session identifier, an identifier identifying a non-web enabled executable application to perform the user command, and a data identifier” as recited in claim 5. Joseph mentions a “value corresponding to an operation protocol,” which is earlier defined as “the protocol to be employed in accessing the second computer system” (Joseph col. 3 lines 64 – 67). The cited passage further merely describes the decomposition of the Uniform Resource Locator to a path and file name, to “provide a resource targeted to the first computer system” from “a resource store that is accessed in accordance with the operation protocol.” As discussed above, Joseph fails to provide enabling disclosure of including “context information and an action request identifier” that is to be acted upon by a non-web enabled executable application, and the Joseph system does not involve the interpretation of any context information that might be included with a URL. Moreover, unlike the present system, Joseph does not disclose use of an “action request identifier” that “identifies at least one of a type of action to be performed by said non-web enabled application, a type of user command, an event, and an authentication service” as recited in claim 5. Consequently, it is respectfully submitted that the rejection of claim 5 is satisfied and should be withdrawn.

Claim 16 provides a method employed by at least one web browser compatible executable application for initiating an action by a non-web enabled executable application. The method includes the activity of generating, in response to a user command, a URL link in a predetermined particular format. The format includes context information and an action request identifier, representing an action to be performed by the non-web enabled executable application. The method also includes the activities of initiating communication of the URL link in a first mode of communication; identifying the particular format link; extracting the context information and action request identifier from the particular format link; and initiating termination of the first mode of communication. The method further includes the activity of initiating communication of the context information and action request identifier to

the second non-web enabled executable application by a second mode of communication different from the first mode of communication.

Claim 16 is considered patentable for the reasons presented above with regard to claim 1. Specifically, because Joseph neither discloses nor suggests “an action request identifier, representing an action to be performed by a non-web enabled executable application” as recited in claim 16, Joseph does not anticipate each element of claim 16. Further, Joseph neither discloses nor suggests “initiating termination of said first mode of communication” and “initiating communication of said context information and action request identifier to said non-web enabled executable application by a second mode of communication different from said first mode of communication” as recited in claim 16. Because the augmented URL that Joseph is transferring is not being handled by a standard browser and represents merely a URL for which the requesting client has no compatible protocol instead of a non-web enabled application, Joseph does not disclose or suggest “initiating termination of said first mode of communication” to stop the browser from transferring to the specified URL. Joseph does not address or mention a “second mode of communication different from said first mode of communication” as recited in claim 16. Finally, Joseph describes only determining “whether the URL contains a first value corresponding to an encapsulating protocol” and “determining whether the [augmented] URL contains an operation protocol key value,” which, contrary to the assertion in the Office Action, are not equivalent to “extracting said context information” from the URL link as claimed in claim 16. Joseph fails to disclose or suggest inclusion or extraction of “context information” in a URL in any manner. Given the simple concatenation of the encapsulating protocol and the operation protocol key to produce the encapsulated URL described in Joseph, and the complete lack of any supporting description of the structure or process of creating the operation protocol key, there is no basis for the assertion that those items include context information that is being “extracted” by the cited operations. Consequently, it is respectfully submitted that the rejection of claim 16 is satisfied and should be withdrawn.

In view of the above remarks and amendments to the claims, it is respectfully requested that this rejection be withdrawn.

Rejection of claims 6-9, 13-14, and 17-19 under 35 U.S.C. 102(e)

Claims 6-9, 13-14, and 17-19 are rejected under 35 U.S.C. 102(e) as being anticipated by Wang et al. (U.S. Patent No. 7,200,683).

Amended independent claim 6 provides a web browser system for receiving and processing application information associated with an action performed by a non-web enabled executable application. The system includes a web browser compatible executable application for receiving a URL link having a particular format via a first mode of communication. The system further identifies the URL link having a particular format, extracts context information and an action request identifier from the particular format URL link, and initiates termination of the first mode of communication used in receiving the particular format URL link (cancelling navigation to the specified URL). The browser application is also employed for initiating communication of the extracted context information and action request identifier received from the web browser compatible executable application to a non-web enabled executable application by a second mode of communication different from the first mode of communication. The system further includes an interface processor for receiving application data from the non-web enabled executable application in response to the action request identifier. The interface processor also parses document object data associated with a predetermined procedure identifier that identifies an executable procedure for processing the received application data to be compatible with the web browser compatible executable application. The interface processor also initiates execution of the identified executable procedure, in response to a command from the interface processor, to provide processed application data to the web browser compatible executable application. Thus, the claimed system enables the communication of appropriate context information and initiation of the execution of a non-web enabled application via the communication processor, and subsequent reception of application data from the non-web enabled application and execution of a procedure to provide the received application data to the web browser compatible executable application in a compatible format via the interface processor. The separation of the communication processor and the interface processor advantageously enables the isolation of the non-web enabled application from the client application, minimizing security concerns that arise from the execution of non-web code from an external source. For the reasons presented below, Applicant respectfully submits that Wang neither discloses nor suggests each feature of amended claim 6.

Wang describes a method for providing graphic user interfaces for controlling devices on a network where the devices provide services (Wang Abstract). The graphic user interfaces of Wang are for display on devices in a first network and on at least one interface device connecting the first network to the second network in which the devices provide services (Wang col. 3 lines 32 – 43). In Wang, the applications supported enable the reading and setting of device parameters to control server devices connected to one network (e.g., a home network) by means of client devices connected to the first network and also connected to another network (e.g., the Internet), for example, a home PC connected to both the Internet and the home network (Wang col. 5 lines 33 – 49). However, Wang does not describe

providing actual processing of communication between web enabled and non-web enabled applications as defined in the present system, i.e., client applications executed by a web browser and accessing a server via the Internet, and non-web enabled applications.

Unlike the claimed system, Wang describes building a browser-based screen page from a GUI description language and a script that uses files such as logos or images representing particular devices so that the logos or images can be displayed as part of the GUI. The GUI functions to enable controlling the devices represented by the logos or images and connected to the home network, according to inputs from the user. However, the Wang system is fundamentally different from and operates in a manner that is not equivalent to the feature combination of claim 6. In Wang, the device to be controlled needs to connect to the home network, for example, via the IEEE 1394 standard or similar standard, and the device on which the GUI is implemented needs to also connect to the home network. The devices being controlled function as servers and the controller device (the PC displaying the GUI) functions as a client (Wang col. 8 lines 21-32), but the interaction between the controlling client and the controlled device is not disclosed to be an equivalent to the communications processor and interface processor of claim 6. Wang does not address processing “a URL link having a particular format” or extracting “context information and an action request identifier from said URL link” as recited in claim 6, because Wang merely obtains the control information from the device in the device-specific form when the GUI is initialized, and then sends back that same data to the device upon user input. Wang sets forth that “each device includes device information such as interface data . . . that provides an interface for commanding and controlling of the device over the network” (Wang col. 5 lines 37 – 41). Thus, Wang neither discloses nor suggests “processing application information” or “extracting context information and an action request identifier from said URL link” as recited in claim 6. There is no processing of such data performed by the Wang client or server system. Instead, Wang merely provides a link in the GUI that activates the device. In contrast, the present system, as recited in claim 6, discloses “receiving and processing application information associated with an action performed by a **non-web enabled** executable application” by the interface processor and further discloses providing “processed received application data to said web browser compatible executable application.” Moreover, Wang does not show or suggest “parsing document object data associated with a predetermined procedure identifier identifying an executable procedure for processing said received application data”. Wang merely cursorily mentions that the environment “supports attaching scripting code to events” (Wang col. 10 lines 38 – 39). Wang is silent regarding the contents of such scripting code or the nature of the events to which scripting code would be attached and provides no disclosure sufficient to reasonably show or suggest the claimed arrangement. Consequently, it is respectfully submitted that the rejection of claim 6 is satisfied and should be withdrawn.

Claims 7 – 9, 13, and 14 are dependent on claim 6 and are considered patentable for the reasons presented above with regard to claim 6.

Amended independent claim 17 provides a method for receiving and processing application information associated with an action performed by an executable application. The method includes the step of receiving a URL link having a particular format via a first mode of communication and identifying the URL link having the particular format. The method also includes the step of extracting context information and an action request identifier from the URL link having the particular format; and initiating termination of the first mode of communication used in receiving the URL link. The method further includes initiating communication of the extracted context information and action request identifier to a non-web enabled executable application by a second mode of communication different from the first mode of communication. The method further includes the step of receiving application data associated with an action performed by a second executable application in response to a user command entered via a displayed browser image. The method further includes parsing document object data associated with a predetermined procedure identifier identifying an executable procedure for processing the received application data to be compatible with a web browser compatible executable application. Finally, the method includes the step of initiating execution of the identified executable procedure to provide processed received application data to the web browser compatible executable application. Applicant respectfully submits that Wang fails to disclose or suggest each feature of amended claim 17.

Claim 17 is considered patentable for the reasons presented above with regard to claims 1 and 6. Although Wang describes controlling devices over a network by using a GUI in a browser based system so that the user can select the device to be controlled and the operation to be performed on it, Wang does not address the operation of the system in terms of processing or parsing data from interaction with a web enabled application. Specifically, Wang neither discloses nor suggests “extracting context information and an action request identifier from said URL link” as recited in claim 17. Wang requires that the devices themselves provide graphical user interface data that allows “displaying a graphical user interface to a user for controlling the devices” (Wang claim 1). The control of the device associated with the graphical user interface data is accomplished when a user selects the GUI element which sends data mirroring what was uploaded to the system on initialization. Thus, Wang does not disclose or suggest “parsing document object data associated with said browser image to find a predetermined procedure identifier identifying an executable procedure for processing said received application data to be compatible with a web browser application” and “initiating execution of said identified executable procedure to provide

processed received information to said web browser application” as recited in claim 17. Wang also does not disclose or suggest processing the “received application data to be compatible with a web browser application” as recited in claim 17 of the present system, but merely passes the data through to the device. Consequently, it is respectfully submitted that the rejection of claim 17 is satisfied and should be withdrawn.

Claims 18 and 19 are dependent on claim 17 and are considered patentable for the reasons presented above.

In view of the above remarks and amendments to the claims, it is respectfully requested that this rejection be withdrawn.

Rejection of claim 2 under 35 U.S.C. 103(a)

Claim 2 is rejected under 35 U.S.C. 103(a) as being unpatentable over Joseph (U.S. Patent No. 6,038,603) in view of Barth et al. (U.S. Patent Application Publication No. 2001/0054020).

Dependent claim 2 provides a system according to claim 1 wherein the first mode of communication comprises at least one of Internet compatible communication and the Hyper Text Transfer Protocol (HTTP), and the second mode of communication comprises at least one of Microsoft Windows compatible message communication, socket communication, and COM (Component Object Model) communication. Claim 2 is dependent on claim 1 and is considered patentable for the reasons presented above with regard to claim 1. Claim 2 is further considered patentable because neither Joseph nor Barth, separately or in combination, discloses or suggests a first mode of communication that comprises “at least one of Internet compatible communication and Hyper Text Transfer Protocol communication” and a second mode of communication that comprises “at least one of Microsoft Windows compatible message communication, socket communication, and COM communication” as recited in claim 2, wherein the system provides for “initiating termination of the first mode of communication” as recited in claim 1.

As previously discussed with regard to claim 1, Joseph neither discloses nor suggests accessing a non-web enabled application through a second mode of communication different from a first mode of communication used to access a web enabled application. Joseph describes using an augmented URL that encapsulates the actual protocol to be used in accessing the desired resource. The proposed encapsulating protocol described in Joseph is “http.” The Office Action asserts that this embodiment satisfies the requirement of a first

mode of communication that comprises at least one of an Internet compatible communication or the Hyper Text Transfer Protocol, and correctly recognizes that Joseph does not support a second mode of communication different from the first. Further, Joseph nowhere discloses or suggests "initiating termination of the first mode of communication" as recited in claim 1 because the URL being transferred in Joseph is an actual URL for a web enabled application whose protocol is not supported by the requesting client application.

Barth describes a method and apparatus for dynamic information collection. The method comprises detecting user actions on a client system, determining that the actions constitute searching for the type of information supported by the system, and then extracting appropriate information from third party web sites, from intermediate databases, and from a set of suppliers who are directly connected to the server, to respond to the user's search (Barth Abstract). In Barth, the user interacts with a server via a web browser using HTTP to access the Internet. The server in Barth then issues requests, possibly without user prompting, to extract information from third party web sites, from intermediate databases, and from direct network connections to suppliers (Barth para. 0030). The communication via direct network connections could be implemented with a second mode of communication from that used for communicating with the Internet, but Barth proposes communicating with the direct network also by use of the HTTP protocol, the same mode of communication Barth uses for the Internet communication (Barth para. 0036 and Figure 9). Barth does disclose elsewhere the use of COM, the Component Object Model, as a communication mode. However, the use of COM that is discussed in Barth is not for communication with a network, but rather to support the monitoring of the client actions on the client machine within the proprietary client browser software that Barth requires all users to install (Barth para. 0163). The proprietary client browser software is used to detect when a client is searching for information that the system can provide (Barth paragraphs 0060 and 0163). Further, Barth (with Joseph), even if suggesting the use of two different modes of communication, fails to disclose "initiating termination of the first mode of communication," because, as in Joseph, the URL does not represent a non-web enabled application, but rather a valid user request to transfer to a valid URL. Also, unlike the present system, which enables users of a web browser compatible application to request and execute applications that are not web enabled and to interact with such applications, Barth merely supplies "boiler plate data display and formatting information to the client" without such data having been explicitly requested by the client (Barth paragraphs 0030 and 0055). Thus, Barth (with Joseph) fails to teach or suggest the use of a "first mode of communication" and a "second mode of communication different from said first mode," wherein the processor utilizing the first mode of communication interprets a URL link received by the first mode of communication and then initiates "termination of the first mode of communication" as claimed in claim 1, to allow the communication of web enabled applications with non-web enabled applications.

Even if Barth were combined with Joseph, the combination fails to suggest the present claimed invention. The combination produces a system in which COM was used for communicating between the portion of the system providing client functions and the portion of the system providing server functions, and a different mode of communication was used for communicating between the server and the other application. This combination does not yield a system wherein “said first mode of communication comprises at least one of Internet compatible communication and Hyper Text Transfer Protocol communication” and “said second mode of communication comprises at least one of Microsoft Windows compatible message communication, socket communication, and COM communication” and the system includes a processor for “initiating termination of said first mode of communication” as recited in claims 1 and 2 of the present system. Additionally, neither Joseph nor Barth, separately or in combination, teaches “initiating termination of a first mode of communication” as recited in claim 1.

Finally, Applicant respectfully submits that there is no reason or motivation to combine Barth with Joseph. Joseph attempts to provide a method and system for communication between applications that do not share a common protocol. Barth deals with the retrieval of information for e-commerce for a user browsing the Internet. The applications with which Barth interfaces are either web enabled applications, or applications under control of the same organization that developed the system disclosed in Barth and thus likely includes customized interfaces for compatible operation. Consequently, both Barth and Joseph begin by working with URLs created by user actions, but they accomplish different objectives with the URLs. Barth need not, and does not, address the problems of handling external application software that might not be compatible with other applications. Barth examines the URL to determine whether the user actions constitute a search for the products Barth supplies, but does not interfere with the normal server handling of the URL to cause transfer to the specified web page. On the other hand, Joseph pre-pends an encapsulating protocol and passes the URL on to another server that can supply the necessary protocol. The proprietary client browser software described by Barth is not relevant to the problem Joseph addresses, specifically, the problem of communication between applications that reside on incompatible servers. Combining Barth with Joseph fails to suggest “initiating termination of said first mode of communication” as recited in claim 1. In view of the above remarks, it is respectfully submitted that the rejection of claim 2 is satisfied and should be withdrawn.

Rejection of claim 3 under 35 U.S.C. 103(a)

Claim 3 is rejected under 35 U.S.C. 103(a) as being unpatentable over Joseph in view of Jasen et al. (U.S. Patent Application Publication No. 2002/0019879).

Amended claim 3 provides a system according to claim 1 wherein the URL link interpreter initiates the termination of the first mode of communication by cancelling navigation to the address indicated in the URL link having the predetermined particular format. Claim 3 is dependent on claim 1 and is considered patentable for the reasons presented above with regard to claim 1. Claim 3 is also considered patentable because neither Joseph nor Jasen, separately or in combination, terminates a first mode of communication by “cancelling navigation to the address indicated in the URL link” as recited in claim 3.

As discussed earlier, Joseph fails to disclose or suggest support of a **non-web enabled** application. Joseph provides a URL Handler communicating with a URL broker that selects and executes an intermediary protocol handler (the resource program) to communicate with an application for which the requesting client has no supporting protocol. Nowhere does Joseph disclose or suggest termination of a first mode of communication by “cancelling navigation to the address indicated in the URL link” as recited in claim 3. In Joseph, the examination of the augmented URL does not result in explicit process termination. The URL is interpreted by the URL Handler that processes all URLs received, both normal URLs and augmented URLs (Joseph Fig. 8 and col. 9 lines 1 – 47).

Jasen provides a method and system for prioritizing network services (Jasen Abstract). The system provides a Network Traffic Management system for prioritizing certain users or certain transactions over others in the allocation of network services (Jasen para. 0018). The system comprises special software (the NTM software) installed on the user's device (Jasen paras. 0019 – 0022). User network traffic or information about the traffic is then routed through the NTM client software to the user's Internet Service Provider (Jasen para. 0022). Although the Office Action asserts that Jasen teaches a system between a network manager and a non-web enabled application, Jasen does not provide action requests or context information to a non-web enabled application. The Jasen system adjusts the level of service provided to an application that is already a web enabled application, and provides that functionality through a system in which there is no “non-web enabled application” except the system itself (Jasen para. 0023). Jasen (with Joseph) nowhere addresses any external application that is non-web enabled beyond a mere mention in the cited paragraph that the prioritization software **could** be used for “other applications including non-Web or non-commerce related applications” (Jasen para. 0018). Jasen is completely silent on how such use might be accomplished or why it would be considered advantageous, and therefore it is non-enabling. Jasen (with Joseph) fails to reasonably disclose or suggest under 35 USC 103 a “system employed by at least one web browser compatible executable application for

initiating an action by a non-web enabled executable application” as recited in claim 1. Moreover, the application in Jasen does not generate “a URL link in a predetermined particular format” “representing an action to be performed by said non-web enabled executable application” as claim 1 further recites. Jasen provides special software that is part of a proprietary package for identifying the characteristics of a network request to determine how it should be prioritized. While the prioritization determination can involve interpretation of a URL, and might include adjustment of the address to which the URL is directed (Jasen para. 0027), it does not include initiating “termination of the first mode of communication by cancelling navigation to the address indicated in said URL” as recited in claim 3.

Additionally, the combination of Jasen with Joseph would not result in a system that “initiates said termination of the first mode of communication by cancelling navigation to the URL” as recited in claim 3. Neither Jasen nor Joseph individually or together disclose or suggest implementation of “cancelling navigation to the address indicated” as recited in claim 3. Consequently, it is respectfully submitted that the rejection of claim 3 is satisfied and should be withdrawn.

Rejection of claims 10, 12, and 15 under 35 U.S.C. 103(a)

Claims 10, 12, and 15 are rejected under 35 U.S.C. 103(a) as being unpatentable over Joseph in view of Wang.

Amended claim 10 provides a system according to claim 6 further including a communication processor coupled to the interface processor for communicating with a **non-web enabled** executable application to acquire the application data. Claim 10 is dependent on claim 6 and is considered patentable for the reasons presented above with respect to claim 6. Specifically, claim 6 recites a URL link interpreter that performs functions of “identifying said URL link . . . extracting context information and an action request identifier . . . and initiating termination of said first mode of communication.” Joseph fails to disclose or suggest handling of “a non-web enabled executable application,” or “extracting context information and an action request identifier” from the URL link, or “initiating termination of said first mode of communication” and communication to a “non-web enabled application by a second mode of communication different from said first mode of communication” as recited in claim 6. Because the application in Joseph is a web enabled application, Joseph need merely identify the appropriate protocol to be used in communicating with the application, rather than “extracting context information and an action request identifier” as recited in claim 6. Because the URL in Joseph represents an actual URL, Joseph does not initiate

“termination of said first mode of communication” as recited in claim 6. Further, although Joseph describes the retrieval of a “resource” from the application whose protocol is not supported, Joseph fails to disclose or suggest any further “processing said received application data” as recited in claim 6, beyond simply using the appropriate protocol to send the retrieved resource to the requesting application. Specifically, Joseph fails to disclose or suggest “initiating execution of said identified executable procedure . . . to provide processed received application data to said browser compatible executable application” as recited in claim 6. Thus, Joseph neither discloses nor suggests each of the features of dependent claim 10.

As previously discussed with regard to claim 6, Wang, similarly to Joseph, fails to disclose or suggest these same features of claim 6. Specifically, Wang fails to disclose or suggest either “initiating termination of said first mode of communication” or “initiating execution of said identifiable executable procedure . . . to provide processed received application data to said browser compatible executable application” as recited in claim 6. The URLs that Wang handles are sent out on the network to the devices that they address, just as would be done with ordinary URLs, without any additional processing by the browser software (Wang col. 5 lines 50 – 60). While Wang describes in some detail the system and method for handling addressing of devices over the network, and creating the GUI dynamically based on the current network configuration, Wang is silent regarding any further processing of the specific control applications after the user interacts with the GUI to request an action. Thus, Wang (with Joseph) neither discloses nor suggests methods of processing the user inputs that involve “initiating termination of said first mode of communication” as recited in claim 6. Although Wang does mention that one of the functions that can be implemented by the described system is “return of information” (Wang col. 5 lines 64 – 65), Wang nowhere discloses or suggests that any processing “to provide processed received application data to said browser compatible executable application,” as recited in claim 6, might be performed on such information. Thus, neither Joseph nor Wang, separately or in combination, teach or suggest “initiating termination of said first mode of communication” or providing “processed received application data to said browser compatible executable application” as recited in claims 6 and 10. Consequently, it is respectfully submitted that the rejection of claim 10 is satisfied and should be withdrawn.

Amended claim 12 provides a system according to claims 6 and 10 wherein the communication processor establishes non-polling communication with the **non-web enabled** executable application, and the non-web enabled executable application provides the status information to the communication processor in response to at least one of a command by the non-web enabled executable application and a request from the communication processor. As recited in claim 12, the present claimed system advantageously enables communication of

status information regarding the non-web enabled application without the requirement of polling by the web client, thereby improving the efficiency of communication between the web client and the non-web enabled application. As correctly recognized in the Office Action, Joseph does not teach establishing “non-polling communication with said non-web enabled executable application and said non-web enabled application provides said status information to said communication processor in response to at least one of a command by said non-web enabled application and a request from said communication processor” as recited in claim 12. Joseph is silent regarding the handling of status information by the portion of Joseph’s system that performs functions analogous to those of the communication processor in the present claimed system.

The Office Action asserts that Wang teaches establishing such communication, and that it would have been obvious to a person of ordinary skill in the art to modify Joseph in view of Wang. However, Wang (with Joseph) does not teach or suggest a system as recited in claim 12 that provides “status information to said communication processor in response to at least one of a command by said non-web enabled executable application and a request from said communication processor.” Instead, Wang specifically discloses the maintaining of an open connection to permit status reporting (col. 10 lines 51 – 57 of Wang recites “the HTTP1.1 specification regarding connection persistence is modified such that an HTTP1.1 client device . . . expects a connection to server device . . . to remain open, because the persistent connection allows full status reporting”). Thus, neither Joseph nor Wang, separately or in combination, disclose or suggest the establishing of non-polling communication “in response to at least one of a command by said non-web enabled executable application and a request from said communication processor” as recited in claim 12. Consequently, it is respectfully submitted that the rejection of claim 12 is satisfied and should be withdrawn.

Amended independent claim 15 provides a system employed by a first web browser compatible executable application for initiating an action by a non-web enabled executable application and for acquiring status information associated with the action. The system includes a URL processor for generating, in response to a user command entered via a displayed browser image, a URL link in a predetermined particular format including context information and an action request identifier. The URL link represents an action to be performed by the non-web enabled executable application. The system also includes a URL link interpreter, coupled to the URL processor via a first mode of communication, identifying said URL link in said predetermined particular format, for extracting said context information and action request identifier from said URL link in said predetermined particular format, and for initiating termination of the first mode of communication. The system further includes a communication processor, coupled to the URL link interpreter, for initiating communication

of the context information and action request identifier to the non-web enabled executable application by a second mode of communication different from the first mode of communication. The system also includes an interface processor for receiving status information associated with the action performed by the non-web enabled executable application in response to the user command, parsing document object data associated with a predetermined procedure identifier identifying an executable procedure for processing the received status information to be compatible with a web browser application; and initiating execution of the identified executable procedure, in response to a command from the interface processor, to provide processed received status information to a web browser application. Applicant respectfully submits that neither Joseph nor Wang, separately or in combination, discloses or suggests the features of claim 15.

Claim 15 is considered patentable for the reasons presented above with respect to similar features of claims 1, 6 and 10. As previously discussed with regard to claims 1 and 10, Joseph does not disclose or suggest the support of "initiating an action by a non-web enabled application" as recited in claim 15. Unlike the present claimed system, Joseph describes support of an application that uses a protocol not supported by a particular client browser and fails to suggest the claimed arrangement involving a non-web enabled application. Joseph also does not teach the generation of a URL link including "context information and an action request identifier" as recited in claim 15. Moreover, as the Office Action correctly recognizes, Joseph fails to teach acquiring information or "receiving status information associated with the action performed by said non-web enabled application" as recited in claim 15. Obviously, then, Joseph further fails to teach "parsing document object data associated with a predetermined procedure identifier identifying an executable procedure for processing said received status information" as recited in claim 15.

The Office Action asserts that Wang discloses acquiring status information and a system employed by a web browser compatible application for receiving status information. However, Wang (with Joseph) does not disclose or suggest "parsing document object data associated with a predetermined procedure identifier identifying an executable procedure for processing said received application data" as recited in claim 15. Wang does not support the actual interoperation of web enabled and non-web enabled applications, but merely a simple return of status information, passing through exactly what it is sent by the device. Thus, Joseph and Wang, separately or in combination, fail to support "parsing document object data" and "processing said received application data" as recited in claim 15. Further, as presented earlier with regard to claim 6, Wang, similarly to Joseph, fails to disclose or suggest "initiating termination of said first mode of communication" as recited in claim 15. Consequently, it is respectfully submitted that the rejection of claim 15 is satisfied and should be withdrawn.

Rejection of claim 11 under 35 U.S.C. 103(a)

Claim 11 is rejected under 35 U.S.C. 103(a) as being unpatentable over Joseph, in view of Wang as applied to claim 10 and in further view of Barth.

Claim 11 provides a system according to claim 10 wherein the communication processor communicates with the non-web enabled executable application by at least one of Microsoft Windows compatible message communication, socket communication, and COM communication. Claim 11 is dependent on claims 6 and 10, and thus arguments presented above with regard to claims 6 and 10 also apply to claim 11.

Joseph, in dealing with non-supported protocols rather than with non-web enabled applications, fails to disclose or suggest communication “with said non-web enabled executable application” as recited in claim 11. Further, as presented above with regard to claims 1 and 10, Joseph (with Wang and Barth) specifically fails to disclose or suggest communication “via a first mode of communication” and “initiating termination of said first mode of communication” as recited in claim 10 upon which claim 11 depends.

Wang does not disclose or suggest the use of any one of the named communication methods for communicating with its non-web enabled devices. Wang contemplates communication via a number of named protocols, e.g., HTTP, UPD, FTP, or other named communication protocols, but is silent regarding the software techniques used for messaging between modules, because Wang does not contemplate “processing said received application data” as recited in claim 6 upon which claims 10 and 11 depend, since Wang does not support any interpretation of communication between applications. Instead, Wang merely proposes a system for retrieving files that are formatted as URLs from the devices to be controlled and then sending the corresponding URL back to the device in response to a user input selecting that URL. Thus, as discussed in detail with regard to claim 10 above, Wang (with Joseph and Barth) specifically fails to disclose or suggest “initiating termination of a first mode of communication” as recited in claim 6, upon which claim 11 depends. Wang does not address a mechanism for generating and later aborting an action request that conforms to the format of a URL to communicate with another application, because Wang merely stores and, upon later selection, returns the URL-format commands that were uploaded to the system by the various devices on initialization. Thus, neither Joseph, nor Wang, nor Barth, separately or in any combination, teach or suggest “initiating termination of said first mode of communication” as recited in claim 6.

Similarly, Barth nowhere contemplates or suggests initiating the termination of said first mode of communication. Barth describes a method and system for determining that a client application is searching for certain product information and providing information that is presumed relevant from a combination of third party web sites, intermediate databases, and a set of suppliers connected by direct network. The system includes proprietary software that runs on the client browser. The system uses the web browser and HTTP to access the internet, and also describes using HTTP to access the direct network. Although Barth does describe using the COM standard, COM is only mentioned as the means of communication between modules of its own proprietary client application software, not for communication with a “non-web enabled executable application” as recited in claim 11. Moreover Barth (with Joseph and Wang) similarly fails to disclose or suggest “initiating termination of the first mode of communication” as recited in claim 6. Thus Barth, alone or in any combination with Joseph and Wang, would still not have provided a system in which the processor initiates termination of the first mode of communication as provided in claims 6, 10, and 11, nor a system in which “said communication processor communicates with said non-web enabled executable application by at least one of Microsoft windows compatible message communication, socket communication, and COM communication” as claim 11 recites. Applicant respectfully submits that these features of claim 11 would not be obvious to one of ordinary skill in the art, since the present claimed system involves using a URL in a non-standard fashion to take advantage of a mechanism provided for control transfers in web browsers. Consequently, it is respectfully submitted that the rejection of claim 11 is satisfied and should be withdrawn.

Rejection of claim 4 under 35 U.S.C. 103(a)

Claim 4 is rejected under 35 U.S.C. 103(a) as being unpatentable over Joseph, in view of Wang, in further view of “The Complete Reference: Internet Millennium edition, copyright 1999 (hereinafter “complete Reference”).

Claim 4 provides a system according to claim 1 in which the user command is received via a displayed web page, and the URL processor sets a location property of a browser window object to an address of the particular format. A user action such as clicking a button on a web page normally generates a URL that is interpreted by a standard browser as an address to which to navigate. The value of the location property of the browser window object is set to the URL generated.

As the Office Action correctly recognizes, Joseph does not teach using a URL “wherein said user command is received via a displayed web page” as recited in claim 4. The Office Action asserts that Wang, however, teaches a displayed web page.

While Wang does teach using a displayed web page to allow specifying functions and actions of a device connected to a network, Wang (with Joseph) fails to address setting “a location property of a browser window object to an address of the particular format” as recited in claim 4. Indeed, Wang is silent on the software techniques used to implement the GUIs for controlling each of the devices in his system. The system in Wang merely uses the browser to act as a conduit for the user interfaces supplied by the devices that are being controlled, receiving the data sent by the devices on initialization, storing it until the corresponding device and function are selected, and then issuing the corresponding URL that the device earlier provided (Wang col. 5 lines 41 – 49).

The Office Action goes on to assert that the Complete Reference teaches setting a browser window object to an address as provided in claim 4. The described action in the Complete Reference is in no way equivalent to setting the location property of a window object to an address, as provided in claim 4. The action described in the Complete Reference is merely that of the user inputting a value into a displayed window field, the address bar, to allow navigating to that URL. This reference does not address the means by which the location property of the browser window is set, or the important concept of the present system that such a value could be interpreted as an action identifier rather than as an actual URL address. The URL processor in the present system “sets a location property of a browser window object to an address of said particular format,” so that its value can be interpreted by the interface processor to determine which non-web enabled application is to be executed, not so that it can be used to navigate to the value specified in the URL as suggested in the Complete Reference. Thus, the claimed arrangement provides for “generating . . . a URL link representing an action to be performed” and “extracting . . . said action request identifier from said URL link in said predetermined particular format” and “initiating termination of the first mode of communication” so that no transfer to the URL specified address actually occurs. Indeed, the Complete Reference citation teaches away from the feature of the claimed system by teaching the normal, expected use of the address bar.

Thus, the combination of Joseph, Wang, and the Complete Reference Internet Millennium Edition fail to disclose or suggest a URL processor that “sets a location property of a browser window object to an address of said particular format” as recited in dependent claim 4 and “a URL link interpreter . . . for initiating termination of the first mode of communication” as further recited in independent claim 1. Consequently, withdrawal of the rejection of claim 4 is respectfully requested.

Having fully addressed Examiner’s rejections, it is believed that, in view of the preceding amendments and remarks, this application stands in condition for allowance. Accordingly, then, reconsideration and allowance are respectfully solicited. If, however, the

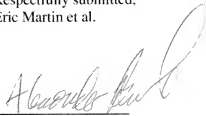
Serial No. 10/826,053

PATENT
03P08954US01

Examiner is of the opinion that such action cannot be taken, the Examiner is invited to contact the Applicant's attorney at the phone number below, so that a mutually convenient date and time for a telephone interview may be scheduled.

Respectfully submitted,
Eric Martin et al.

By:



Alexander J. Burke
Reg. No. 40,425

June 9, 2008

Siemens Corporation
Customer No. 28524
Tel 732 321 3023
Fax 732 321 3030
June 9, 2008